

Утверждаю
Председатель Новочеркасского
территориального Совета директоров
учреждений профессионального
образования
И. А. Потапов
« 15 » 2018 г.



Положение

о проведении личной интернет - олимпиады по программированию среди студентов учреждений профессионального образования Новочеркасского территориального объединения

1 Общие положения

1.1. Интернет-олимпиада по программированию среди студентов учреждений профессионального образования проводится Советом директоров Новочеркасского территориального объединения.

1.2. Организаторами олимпиады являются Совет директоров Новочеркасского территориального объединения и территориальное методическое объединение преподавателей информатики ССУЗов.

1.3. К участию в интернет-олимпиаде приглашаются все желающие студенты учреждений профессионального образования Новочеркасского территориального объединения.

1.4. Интернет-олимпиада представляет собой выполнение конкретных заданий различной сложности, с учетом времени выполнения и других критериев, проводимые в течение определенного периода и завершающиеся церемонией чествования победителей.

2 Цели и задачи олимпиады

2.1. Основными целями и задачами интернет-олимпиады являются пропаганда научных знаний и развитие у студентов интереса к научной деятельности, активизация работы спецкурсов, кружков, научных обществ учащихся, развития других форм работы со студентами, создание оптимальных условий для выявления одаренных и талантливых студентов, их дальнейшего интеллектуального развития и профессиональной ориентации, развитие умения работать в коллективе.

3 Условия и порядок проведения интернет-олимпиады

3.1. В Олимпиаде по программированию студентам предлагается решить **20** задач разной сложности. Продолжительность тура — **12 дней**. Начало олимпиады **01.12.18 с 09:00**. Окончание турнира **12.12.18 в 23:00**. Система проверки будет работать круглосуточно. Жюри может продлить время тура в случае каких-либо непредвиденных обстоятельств. Участники используют современные компьютеры с установленной операционной системой семейства Windows.

3.2. Для решения задач участникам предлагается использовать следующие языки программирования и соответствующие им среды разработки:

- ✓ Pascal: Borland Pascal 7.0, Borland Delphi 7.0, Free Pascal 2.4.4;
- ✓ C/C++: Borland C++ 3.1, Visual C++ 2010 Express Edition, GNU C++ 4.6.1 (MinGW);
- ✓ Java: Sun JDK 1.6.0_27, Eclipse 3.6.
- ✓ Visual C# 2010 Express Edition.

Разрешается использовать и другие среды для выше указанных языков программирования.

3.3. Во время тура участники решают предложенные задачи, разрешается пользоваться любой литературой и личными записями. Решением задачи является программа, составленная на одном из допустимых языков программирования. Программа не должна

содержать вспомогательных модулей или файлов. Разные задачи можно решать на разных языках программирования. За решение одной и той же задачи более чем на одном языке программирования дополнительные баллы не начисляются.

Все задачи предполагают, что входные данные вводятся с клавиатуры, а выходные данные выводятся на экран.

3.4. Проверка решений производится во время соревнований. Решения проверяются системой CONTESTER, поэтому: нужно будет зарегистрироваться в этой системе (один раз), входить в систему под своим именем, решать задачи и отправлять на проверку в этой системе. CONTESTER проверит решение и сообщит о правильности написания программы.

3.5. Для проведения тестирования программ Вам необходимо: для работы в контестере перейти по адресу <http://contestik.ru> и выбрать в открывшемся меню **Главная** под блоком «Регистрация» ссылку «Зарегистрироваться». Перед вами откроется форма **Регистрация**. Заполните поля, помеченные звездочкой. **В поле «Имя» обязательно запишите по-русски через пробел сокращенное названия учреждения (где учитесь), фамилию, имя и отчество полностью.** После регистрации необходимо войти в систему, указав имя и пароль.

3.6. Для отправки программы на проверку и просмотра результатов соревнований нужно: в левой части страницы (Первая интернет-олимпиада по программированию) имеется ссылка «**Турнирная таблица**», которая позволит посмотреть текущие результаты соревнования. Для отправки программы на проверку необходимо скопировать в буфер обмена код написанной программы и вставить его в окно «**Текст попытки**» или, используя ссылку **файл попытки**, выберите файл, который хотите сдать на проверку. Далее из списка «**Компилятор**» выбираете язык программирования, на котором была написана программа и нажмите на кнопку «**Отправить**». В меню системы CONTESTER можно посмотреть, как оформлять решения на различных языках программирования. Решения участников проверяются на заранее подготовленном жюри наборе тестов. Решение принимается, если оно прошло все тесты. Частичные решения (прошедшие не все тесты) не оцениваются. Если на каком-то тесте произошла ошибка, то сообщается только номер теста и код ошибки (общее количество тестов для данной задачи не сообщается). Время тестирования и решения и доступная память на каждом тесте ограничены. Решения, превысившие установленное ограничение, считаются неэффективными для данной задачи. В этом случае тест считается не пройденным, а решение, как следствие, неверным. В контестере указывается ограничения на время работы программы на одном тесте и на размер доступной памяти в формулировках задач.

3.7. В решениях задач запрещено использовать:

- расширенную память и защищенный режим процессора при использовании 16-битных компиляторов;
- чтение и запись векторов прерываний;
- создание подкаталогов;
- любое использование сетевых средств;
- любые другие средства или действия, которые могут нарушить процесс проверки и прохождения соревнований.

4 Система оценивания и определения победителей

4.1 В таблице результатов выше оказывается тот, кто решит большее количество задач. При равенстве количества решенных задач выше оказывается тот, у которого меньше суммарное штрафное время. Штрафное время вычисляется следующим образом: для решенных задач, штрафное время представляет собой время в минутах, прошедшее с начало тура до сдачи задачи, плюс 20 штрафных минут за каждую неудачную попытку сдать задачу. За нерешенные задачи штрафное время не начисляется.

4.2 Апелляция не проводится. Жюри обладает исключительным правом определения правильности прохождения тестов, выставления оценок, определения победителей и

дисквалификации участников. Жюри разбирает вопросы, возникающие в результате непредвиденных событий и обстоятельств. Решения жюри окончательны и обжалованию не подлежат.

5 Ответственные за проведение интернет-олимпиады

Морозова Татьяна Игоревна, председатель ТМО преподавателей информатики (89081782576)

Твердохлебова Татьяна Григорьевна преподаватель (89045009930)

Филиппенко Георгий Георгиевич (89045081726)

Требование к программному продукту

Программа не должна содержать приглашения к вводу

Как не должно быть	Как должно быть
Write('N?'); Read(N);	Read(N);
cout <<"N?"; cin >>N;	cin >>N;

Программа не должна содержать подключение библиотек, предназначенных для управления экраном

Этого должно быть
Uses CRT;
#include <conio.h>

Выводить данные в требуемом формате

Например, распечатать массив целых чисел из N (N<101) элементов в строку через пробел

Как не должно быть	Как должно быть
Var A:arra[1..100] of integer; i,N: Integer; begin read(N); for i:=1 to N do Read(a[i]); for i:=1 to N do Write(a[i]); end.	Var A:arra[1..100] of integer; i,N: Integer; begin read(N); for i:=1 to N do Read(a[i]); for i:=1 to N do Write(a[i], ' '); end.
Var A:arra[1..100] of real; i,N: Integer; begin read(N); for i:=1 to N do Read(a[i]); for i:=1 to N do Write(a[i], ' '); end.	Var A:arra[1..100] of real; i,N: Integer; begin read(N); for i:=1 to N do Read(a[i]); for i:=1 to N do Write(a[i]:0:0, ' '); end.
#include <iostream> using namespace std; int main()	#include <iostream> using namespace std; int main()

<pre> { int N,i,A[100]; cin>>N; for (i=0;i<N; i++) cin>>A[i]; for (i=0;i<N; i++) cout<<A[i]; return 0; } </pre>	<pre> { int N,i,A[100]; cin>>N; for (i=0;i<N; i++) cin>>A[i]; for (i=0;i<N; i++) cout<<A[i]<<' '; return 0; } </pre>
<pre> #include <iostream> using namespace std; int main() { int N,I; float A[100]; cin>>N; for (i=0;i<N; i++) cin>>A[i]; for (i=0;i<N; i++) cout<<A[i]<<' '; return 0; } </pre>	<pre> #include <iostream> #include <stdio.h> using namespace std; int main() { int N,I; float A[100]; cin>>N; for (i=0;i<N; i++) cin>>A[i]; for (i=0;i<N; i++) printf("%0.0f ",A[i]; return 0; } </pre>

PascalABC часто выводит вещественные числа в виде целого числа, а по стандарту число будет выведено в плавающей форме! – САМАЯ ПОПУЛЯРНАЯ ОШИБКА!

Не делать задержек после вывода

Например, распечатать массив целых чисел из N (N<101) элементов в строку через пробел

Как не должно быть	Как должно быть
<pre> Var A:arra[1..100] of integer; i,N: Integer; begin read(N); for i:=1 to N do Read(a[i]); for i:=1 to N do Write(a[i], ' '); Readln; Readln; end. </pre>	<pre> Var A:arra[1..100] of integer; i,N: Integer; begin read(N); for i:=1 to N do Read(a[i]); for i:=1 to N do Write(a[i], ' '); end. </pre>
<pre> Var A:arra[1..100] of real; i,N: Integer; begin read(N); for i:=1 to N do Read(a[i]); for i:=1 to N do Write(a[i], ' '); end. </pre>	<pre> Var A:arra[1..100] of real; i,N: Integer; begin read(N); for i:=1 to N do Read(a[i]); for i:=1 to N do Write(a[i]:0:0, ' '); end. </pre>

Отправка решения программ в Visual Studio

Программа должна сдаваться в контестер без первой строки

```
#include <stdafx>
```

Пример применения математических функций на C

```
#include "cstdlib"
#include "math.h"// математическая библиотека
#include "stdio.h"
using namespace std;

int main()
{
int a1, b1, a2, b2, S1, S2, k;
scanf("%d %d %d %d",&a1,&b1,&a2,&b2);
S1=a1*b1; S2=a2*b2;
k=abs(S2-S1); // для вещественных переменных fabs
printf("%d",k);
return 0;
}
```

Пример вывода с заданной точностью

Вывод двух чисел через пробел с точностью до 3-х знаков после точки

Пример на C

```
...
float a;
double b;
....
printf("%0.3f %0.3lf",a,b);
```

Пример на Паскале

```
...
Var
A,B:real;
...
Write(A:0:3, ' ',B:0:3);
```